This Page Is Inserted by IFW Operations and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents will not correct images, please do not report the images to the Image Problem Mailbox.

WEST Search History

	200000000000000000000000000000000000000	500000000000000000000000000000000000000	300000000000000000000000000000000000000
Hide Items		2000	2000
- Into Itomo	- Wastara		- Anan

	200000000000000000000000000000000000000	200000000000000000000000000000000000000	- 8 000000000000000000000000000000000000
		500000000000000000000000000000000000000	500000000000000000000000000000000000000

DATE: Saturday, July 17, 2004

Hide? <u>Set Name</u> <u>Query</u>		Hit Count	
	DB=PG	PB,USPT,USOC; PLUR=YES; OP=ADJ	
	L9	(L6 or I7 or I8) and graph	3
	L8	L3 and sigma	2
	L7	L3 and priority	21
	L6	L3 and severity	3
	L5	L3 and I1	5
DB=EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ			
	L4	L3	9
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ			
	L3	bug near2 (tracking or tracker or rating or evaluat\$)	78
	L2	bug near2 (tracking or tracker)	59
DB=PGPB,USPT,USOC; PLUR=YES; OP=ADJ			
	L1	717/124-133.ccls. or 714/15-18,20-21,47-54.ccls.	4504

END OF SEARCH HISTORY



Membership Publications/Services Standards Conferences Careers/Jobs IBBB XNOW

Welcome



	United States Patent and Trademark Office Million D: Million D:
Help FAQ Terms IE	EE Peer Review Quick Links ** Search Re
Welcome to IEEE Xplore*	
O- Home O- What Can I Access? O- Log-out	Your search matched 10 of 1051129 documents. A maximum of 500 results are displayed, 15 to a page, sorted by Relevance in Descending order.
Tables of Contents O- Journals	Refine This Search: You may refine your search by editing the current search expression or entering a new one in the text box.
& Magazines — Conference Proceedings	tracking <and>bug Check to search within this result set</and>
O- Standards Search	Results Key: JNL = Journal or Magazine CNF = Conference STD = Standard
O- By Author O- Basic O- Advanced Member Services	1 Automated bug tracking: the promise and the pitfalls McLaughlin, L.; Software, IEEE, Volume: 21, Issue: 1, JanFeb. 2004 Pages: 100 - 103
O- Join IEEE O- Establish IEEE	[Abstract] [PDF Full-Text (182 KB)] IEEE JNL
Web Account O Access the IEEE Member Digital Library O Access the	2 Populating a Release History Database from version control and bug tracking systems Fischer, M.; Pinzger, M.; Gall, H.; Software Maintenance, 2003. ICSM 2003. Proceedings. International Conference on , 22-26 Sept. 2003 Pages:23 - 32
LEEE Enterprise File Cabinet	[Abstract] [PDF Full-Text (336 KB)] IEEE CNF
Print Format	3 Mining the maintenance history of a legacy software system Jelber Sayyad Shirabad; Lethbridge, T.C.; Matwin, S.; Software Maintenance, 2003. ICSM 2003. Proceedings. International Conference

[Abstract] [PDF Full-Text (337 KB)] IEEE CNF

4 Analyzing and relating bug report data for feature tracking

Fischer, M.; Pinzger, M.; Gall, H.;

on, 22-26 Sept. 2003

Pages:95 - 104

Reverse Engineering, 2003. WCRE 2003. Proceedings. 10th Working Conference on, 13-16 Nov. 2003

Pages:90 - 99

[Abstract] [PDF Full-Text (797 KB)] IEEE CNF

5 Tracking down software bugs using automatic anomaly detection

Hangal, S.; Lam, M.S.;

Software Engineering, 2002. ICSE 2002. Proceedings of the 24rd International Conference on , 19-25 May 2002

Pages:291 - 301

[Abstract] [PDF Full-Text (1285 KB)] **IEEE CNF**

6 Issue tracking

Johnson, J.N.; Dubois, P.F.;

Computing in Science & Engineering [see also IEEE Computational Science and Engineering], Volume: 5, Issue: 6, Nov.-Dec. 2003

Pages:71 - 77

[Abstract] [PDF Full-Text (1430 KB)] **IEEE JNL**

7 Seesoft-a tool for visualizing line oriented software statistics

Eick, S.C.; Steffen, J.L.; Sumner, E.E., Jr.;

Software Engineering, IEEE Transactions on , Volume: 18 , Issue: 11 , Nov. 1992

Pages:957 - 968

[Abstract] [PDF Full-Text (668 KB)] **IEEE JNL**

8 Interactive static analysis of Ada programs

Vanek, L.; Gogan, V.; Culp, M.; Berkowitz, S.;

Digital Avionics Systems Conference, 1990. Proceedings., IEEE/AIAA/NASA

9th, 15-18 Oct. 1990

Pages:165 - 170

[Abstract] [PDF Full-Text (488 KB)] **IEEE CNF**

9 Static analysis of program source code using EDSA

Vanek, L.I.; Culp, M.N.;

Software Maintenance, 1989., Proceedings., Conference on , 16-19 Oct. 1989

Pages:192 - 199

[Abstract] [PDF Full-Text (604 KB)] IEEE CNF

10 A knowledge-based approach for real-time systems debugging

Tsai, J.P.; Fang, K.Y.; Thalla, V.R.K.; Gandhi, H.;

System Sciences, 1988. Vol.II. Software Track, Proceedings of the Twenty-First Annual Hawaii International Conference on , Volume: 2 , 5-8 Jan. 1988

Pages:533 - 540

[Abstract] [PDF Full-Text (576 KB)]

Home | Log-out | Journals | Conference Proceedings | Standards | Search by Author | Basic Search | Advanced Search | Join IEEE | Web Account | New this week | OPAC Linking Information | Your Feedback | Technical Support | Email Alerting | No Robots Please | Release Notes | IEEE Online Publications | Help | FAQ| Terms | Back to Top

Copyright © 2004 IEEE — All rights reserved

US Patent & Trademark Office

Subscribe (Full Service) Register (Limited Service, Free) Login

Search:

The ACM Digital Library O The Guide

+bug +tracking +severity

January 1

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used bug tracking severity

Found 59 of 139.567

Sort results by

relevance

Save results to a Binder ? Search Tips

Try an Advanced Search Try this search in The ACM Guide

Display results

expanded form

Open results in a new window

Results 1 - 20 of 59

Result page: 1 2 3

next

Relevance scale

The whiteboard: Tracking usability issues: to bug or not to bug?

May 2001 interactions, Volume 8 Issue 3

Full text available: pdf(162.71 KB) [3] html(22.19 KB)

Additional Information: full citation, references, citings, index terms

2 Analyzing and communicating usability data: now that you have the data what do you do? a CHI'94 workshop



Nandini P. Nayak, Debbie Mrazek, David R. Smith January 1995 ACM SIGCHI Bulletin, Volume 27 Issue 1

Full text available: pdf(999.36 KB) Additional Information: full citation, abstract, citings, index terms

At the CHI '94 Workshop: "Analyzing and Communicating Usability Data", usability researchers and practitioners gathered to discuss how usability data should be collected. analyzed, and communicated to help ensure the design and development of usable products. This workshop was motivated by the challenges CHI practitioners have encountered in analyzing and communicating data despite the widespread availability of a variety of usability tools. Workshop activities included brief presentations, smal ...

Functional verification of large ASICs

Adrian Evans, Allan Silburt, Gary Vrckovnik, Thane Brown, Mario Dufresne, Geoffrey Hall, Tung Ho. Yina Liu

May 1998 Proceedings of the 35th annual conference on Design automation - Volume

Full text available: pdf(263.28 KB)

Publisher Site

Additional Information: full citation, abstract, references, citings, index terms

This paper describes the functional verification effort during a specific hardware development program that included three of the largest ASICs designed at Nortel. These devices marked a transition point in methodology as verification took front and centre on the critical path of the ASIC schedule. Both the simulation and emulation strategies are presented. The simulation methodology introduced new techniques such as ASIC subsystem level behavioural modeling, large multi-chip simulations, ...

Keywords: ASIC verification, emulation, simulation

Tracking pointers with path and context sensitivity for bug detection in C programs V. Benjamin Livshits, Monica S. Lam

September 2003 ACM SIGSOFT Software Engineering Notes, Proceedings of the 9th European software engineering conference held jointly with 10th ACM SIGSOFT international symposium on Foundations of software

engineering, Volume 28 Issue 5

Full text available: 📆 pdf(220.74 KB) Additional Information: full citation, abstract, references, index terms

This paper proposes a pointer alias analysis for automatic error detection. State-of-the-art pointer alias analyses are either too slow or too imprecise for finding errors in real-life programs. We propose a hybrid pointer analysis that tracks actively manipulated pointers held in local variables and parameters accurately with path and context sensitivity and handles pointers stored in recursive data structures less precisely but efficiently. We make the unsound assumption that pointers passed i ...

Keywords: SSA representation, buffer overruns, context-sensitive analysis, error detection, path-sensitive analysis, pointer analysis, program analysis, program representation, security flaws, software security

5 Conscientious programming using PMA

Guy Barker, Douglas J. Keenan, Herman van Loon

May 1990 ACM SIGAPL APL Quote Quad, Conference proceedings on APL 90: for the future, Volume 20 Issue 4

Additional Information: full citation, abstract, references, citings, index Full text available: pdf(773.08 KB) terms, review

This paper describes the use of APL systems at Philips to assist in providing a controlled application support environment and examines in detail an APL software management system, PMA (Program Maintenance Aid).

A system and language for building system-specific, static analyses Seth Hallem, Benjamin Chelf, Yichen Xie, Dawson Engler May 2002 ACM SIGPLAN Notices, Proceedings of the ACM SIGPLAN 2002 Conference on Programming language design and implementation, Volume 37 Issue 5

Full text available: pdf(276.96 KB)

Additional Information: full citation, abstract, references, citings, index terms

This paper presents a novel approach to bug-finding analysis and an implementation of that approach. Our goal is to find as many serious bugs as possible. To do so, we designed a flexible, easy-to-use extension language for specifying analyses and an efficent algorithm for executing these extensions. The language, metal, allows the users of our system to specify a broad class of analyses in terms that resemble the intuitive description of the rules that they check. The system, xqcc ...

Keywords: error detection, extensible compilation

7 Simulation coverage and generation for verification: Coverage-oriented verification of banias

Alon Gluska

June 2003 Proceedings of the 40th conference on Design automation

Full text available: pdf(178.40 KB) Additional Information: full citation, abstract, references, index terms

The growing complexity of state-of-art microprocessors dictates the use of cost-effective verification methods. Functional coverage was widely applied in the verification of Banias, Intel's new IA-32 microprocessor designed solely for the mobile computing market. In this paper, we describe the practical coverage approach as was carried out in the verification of Banias. According to this Coverage-Oriented verification approach, focus shifts gradually from basic logic cleanup using random testing ...

Keywords: coverage, functional coverage, logic design, logic verification

8 Technical papers: concurrency: Software model checking in practice: an industrial case study

Satish Chandra, Patrice Godefroid, Christopher Palm

May 2002 Proceedings of the 24th international conference on Software engineering

Full text available: pdf(1.16 MB) Additional Information: full citation, abstract, references, index terms

We present an application of software model checking to the analysis of a large industrial software product: Lucent Technologies' CDMA call-processing library. This software is deployed on thousands of base stations in wireless networks world-wide, where it sets up and manages millions of calls to and from mobile devices everyday. Our analysis of this software was carried out using VeriSoft, a tool developed at Bell Laboratories that implements model-checking algorithms for systematically testin ...

9 Article abstracts with full text online: A study to investigate the impact of requirements instability on software defects



Talha Javed, Manzil e Magsood, Qaiser S. Durrani

May 2004 ACM SIGSOFT Software Engineering Notes, Volume 29 Issue 3

Full text available: pdf(353.10 KB) Additional Information: full citation, abstract, references

Software development is a dynamic process and is characterized by change. Software projects often begin with unclear, ambiguous, and incomplete requirements which give rise to intrinsic volatility. Constant change in requirements is one of the main causes of software defects and a major issue faced by the software industry. This paper describes the findings of our research-based study that investigates the impact of both the pre-release and postrelease requirements changes on overall defects by ...

Keywords: change request (CR's)1, defects, high/medium/low change requests, pre/post release changes, requirements change, severity-1/severity-2 defects

10 Session 3: Characterizing modes of coordination: a comparison between oral and artifact based coordination



Peter H. Carstensen, Morten Nielsen

September 2001 Proceedings of the 2001 International ACM SIGGROUP Conference on **Supporting Group Work**

Full text available: pdf(473.05 KB) Additional Information: full citation, abstract, references, index terms

The choice of communicative modality will greatly affect the way cooperative work is coordinated. Computer supported coordination brings about changes to communicative modalities-often the change is from oral to artifact based coordination. In order to inform the designed changes in modality we need to understand the characteristics of individual modes of coordination, and we need to compare modes before changes are implemented. Within this context the paper has two objectives: (1) to characteri ...

Keywords: coordination modes, maritime operations, software testing

11 Two case studies of open source software development: Apache and Mozilla July 2002 ACM Transactions on Software Engineering and Methodology (TOSEM),



Full text available: pdf(373.10 KB)

Additional Information: full citation, abstract, references, index terms, review

According to its proponents, open source style software development has the capacity to compete successfully, and perhaps in many cases displace, traditional commercial development methods. In order to begin investigating such claims, we examine data from two major open source projects, the Apache web server and the Mozilla browser. By using email archives of source code change history and problem reports we quantify aspects of developer participation, core team size, code ownership, productivit ...

Keywords: Apache, Mozilla, Open source software, code ownership, defect density, repair interval

12 On the contributions of different empirical data in usability testing

Maria R. Ebling, Bonnie E. John

August 2000 Proceedings of the conference on Designing interactive systems: processes, practices, methods, and techniques

Full text available: pdf(281.55 KB)

Additional Information: full citation, abstract, references, citings, index terms

Many sources of empirical data can be used to evaluate an interface (e.g., time to learn, time to perform benchmark tasks, number of errors on benchmark tasks, answers on questionnaires, comments made in verbal protocols). This paper examines the relative contributions of both quantidtadtive and qualitative data gathered during a usability study. For each usability problem uncovered by this study, we trace each contributing piece of evidence back to its empirical source. For this usability ...

Keywords: empirical data, usability testing, verbal protocol

13 Token-based scanning of source code for security problems

John Viega, J. T. Bloch, Tadayoshi Kohno, Gary McGraw

August 2002 ACM Transactions on Information and System Security (TISSEC), Volume 5

Full text available: 1 pdf(221.51 KB) Additional Information: full citation, abstract, references, index terms

We describe **ITS4**, a tool for statically scanning C and C++ source code for security vulnerabilities. Compared to other approaches, our scanning technique stakes out a new middle ground between accuracy and efficiency. This method is efficient enough to offer real-time feedback to developers during coding while producing few false negatives. Unlike other techniques, our method is also simple enough to scan C++ code despite the complexities inherent in the language. Using ITS4, we fo ...

Keywords: Buffer overflows, race conditions, security analysis

14 An empirical study of global software development: distance and speed James D. Herbsleb, Audris Mockus, Thomas A. Finholt, Rebecca E. Grinter July 2001 Proceedings of the 23rd international conference on Software engineering

Full text available: pdf(151.47 KB) Additional Information: full citation, abstract, references, citings, index terms

Global software development is rapidly becoming the norm for technology companies. Previous qualitative research suggests that multi-site development may increase development cycle time. We use both survey data and data from the source code change management system to model the extent of delay in a multi-site software development organization, and explore several possible mechanisms for this delay. We also measure differences in same-site and cross-site communication patterns, and analyze the ...

Keywords: awareness, delay, global collaboration, informal communication, software development, speed

15 <u>Security and privacy: Securing web application code by static analysis and runtime protection</u>



Yao-Wen Huang, Fang Yu, Christian Hang, Chung-Hung Tsai, Der-Tsai Lee, Sy-Yen Kuo May 2004 Proceedings of the 13th conference on World Wide Web

Full text available: pdf(608.77 KB) Additional Information: full citation, abstract, references, index terms

Security remains a major roadblock to universal acceptance of the Web for many kinds of transactions, especially since the recent sharp increase in remotely exploitable vulnerabilities have been attributed to Web application bugs. Many verification tools are discovering previously unknown vulnerabilities in legacy C programs, raising hopes that the same success can be achieved with Web applications. In this paper, we describe a sound and holistic approach to ensuring Web application security. Vi ...

Keywords: information flow, noninterference, program security, security vulnerabilities, type systems, verification, web application security

¹⁶ A Modification Request Control System

D. B. Knudsen, A. Barofsky, L. R. Satz

October 1976 Proceedings of the 2nd international conference on Software engineering

Full text available: pdf(666.80 KB)

Additional Information: full citation, abstract, references, citings, index terms

The Modification Request Control System (MRCS) tracks and reports project change requests and resulting activity through interactive input and extraction of change request data from computer files. MRCS is one of the tools available as part of the Programmer's Workbench (PWB). It was developed to aid in the timely control and coordination of software changes. It provides the capability to: (1) interactively create, update, and print MRS; (2) track and record the flow of the MR through the s ...

Keywords: Project management, Software control, UNIX

17 PROUST: Knowledge-based program understanding

W. Lewis Johnson, Elliot Soloway

March 1984 Proceedings of the 7th international conference on Software engineering

Full text available: pdf(784.73 KB)

Additional Information: full citation, abstract, references, citings, index terms

This paper1 describes a program called PROUST which does on- line analysis and understanding of Pascal programs written by novice programmers. PROUST takes as input a program and a non-algorithmic description of the program requirements, and finds the most likely mapping-between the requirements and the code. This mapping is in essence a reconstruction of the design and implementation steps that the programmer went through in writing the program. A knowledge base of pro ...

18 Of maps and scripts—the status of formal constructs in cooperative work Kjeld Schmidt



November 1997 Proceedings of the international ACM SIGGROUP conference on Supporting group work: the integration challenge: the integration challenge

Full text available: pdf(1.64 MB)

Additional Information: full citation, references, citings, index terms

19 Object lens: a "spreadsheet" for cooperative work

Kum-Yew Lai, Thomas W. Malone

January 1988 Proceedings of the 1988 ACM conference on Computer-supported cooperative work

Full text available: pdf(959.70 KB)

Additional Information: full citation, abstract, references, citings, index terms

Object Lens allows unsophisticated computer users to create their own cooperative work applications using a set of simple, but powerful, building blocks. By defining and modifying templates for various semistructured objects, users represent many different kinds of information. By creating semiautonomous agents, users specify rules for automatically processing this information in different situations. The combination of these primitives provides ...

²⁰ Interface design issues for advice-giving expert systems

John M. Carroll, Jean McKendree

January 1987 Communications of the ACM, Volume 30 Issue 1

Full text available: pdf(2.28 MB)

Additional Information: full citation, abstract, references, citings, index terms

Advice giving could become the first successful domain for intelligent interfaces.

Results 1 - 20 of 59

Result page: $1 \quad 2 \quad 3$

next

The ACM Portal is published by the Association for Computing Machinery. Copyright @ 2004 ACM, Inc. Terms of Usage Privacy Policy Code of Ethics Contact Us

Useful downloads: Adobe Acrobat QuickTime Windows Media Player